

Contents

1	OSX developer install	1
1.1	Install Python	1
1.2	Test your python	2
1.3	Install PmagPy	3
1.4	Test PmagPy	4
1.5	Finishing up	4
1.6	Keeping PmagPy up-to-date	4
2	Next steps	5

1 OSX developer install

These are the install instructions for OSX/developer. If this isn't your operating system, or you want a developer install, go back to the Cookbook and select the correct link.

1.1 Install Python

First, you will need to download a scientific Python distribution. We strongly recommend Anaconda. Be warned, your computer comes with a version of Python already installed; but this pre-installed version does NOT have everything you will need to run **PmagPy**, so you will still need to download Anaconda.

- Download and install Anaconda Python 3.
- Note: if you already installed Anaconda Python, and then you upgrade to OSX Catalina, you may need to reinstall Anaconda. In general, if you run into errors with your Anaconda Python environment, it can save time to simply uninstall and reinstall.
- Open your command line (Terminal). See this Cookbook section for more information on finding your command line.
- something happened in recent versions of pandas that breaks certain programs, so you need to downgrade your pandas installation:
conda install pandas version==1.2.5
- Create and activate a new conda Python environment with some required packages:

```
conda create -n pmagpy_env future wxPython cartopy pandas matplotlib request
conda activate pmagpy_env
pip install --user --upgrade pip setuptools
conda install conda-forge::python-wget
```

- You have now created a new environment called “pmagpy_env” and activated it. This gives you a Python environment with the packages you need to run PmagPy programs. Each time you want to use PmagPy, you will open your Terminal and then run:

```
conda activate pmagpy_env
```

You can also deactivate the environment using:

```
conda deactivate
```

You must always activate your pmagpy environment when you open a new Terminal window if you want to use PmagPy in that window. To learn more about managing conda environments, etc., see the Anaconda documentation and cheatsheet.

- When running these commands you may see some warnings about dependencies, but you can just ignore them!
- If a package fails to install, you may need to use “sudo”: i.e., `sudo pip install scripttest`. You will then be asked for your computer password.

1.2 Test your python

To make sure that you have installed Python successfully, type `python` on your command line. You should see something like this:

```
Python 3.7.10 (default, Feb 26 2021, 10:16:00)
[Clang 10.0.0 ] :: Anaconda, Inc. on darwin
Type "help", "copyright", "credits" or "license" for more information.
>>>
```

(Press control-D to exit)

1.3 Install PmagPy

Here are the steps to clone and install **PmagPy**.

- **PmagPy** has been most extensively tested using the bash shell (not csh, zsh, etc.). However, bash is no longer the default for OSX. The new default shell (zsh) should work, but if you run into trouble and need to switch, you can follow these instructions. Select Terminal --> Preferences --> General, and choose “default login shell”. Restart Terminal, and you’ll be ready to go.
- Download and install git
- Navigate to the directory where you want to put the PmagPy folder, then run:

```
git clone https://github.com/PmagPy/PmagPy.git
```

- You should now have a full local copy of the PmagPy repository. Change directories into PmagPy:

```
cd PmagPy
```

- Next, you need to add PmagPy to your PATH so that the PmagPy programs can be called from any directory. Try running:

```
python dev_setup.py install
```

- If you have problems with the install, run `python dev_setup.py -h` for more information. You can also set your PATH manually if `dev_setup.py` fails.
- After completing the developer install, you should restart your command line.

1.4 Test PmagPy

Test core functionality:

- On your command line, type “python” to start the interpreter. You will import pmagpy and then run a simple calculation. Importing pmagpy may take over a minute the first time, so be patient! It will be much faster in subsequent runs.

```
>>> from pmagpy import pmag
>>> pmag.angle([350.0,10.0],[320.0,20.0])
array([30.59060998])
>>>
```

Test the GUIs:

- On the command line, open Pmag GUI by running:

```
pmag_gui.py
```

Remember that the program may be very slow to initialize the first time! You may also need to resize the GUI window. If the program doesn’t open or you get an error message, go back and carefully follow the install instructions to make sure you didn’t miss a step. If you still have a problem, try the Troubleshooting section. If you don’t find an answer there, check out the existing Github issues and create a new one if necessary.

1.5 Finishing up

Accessing example data files:

There are many data files used in the examples of programs and for use with the textbook Essentials of Paleomagnetism. You can find these in PmagPy/data_files.

1.6 Keeping PmagPy up-to-date

You will want to stay up to date with **PmagPy** development. To update your developer install, you will just need to navigate to the **PmagPy** directory and run:

```
git pull
```

This will grab all of the latest code from Github, and will be immediately available to you.

2 Next steps

Back to the Cookbook!